

## PARTICLE FILTERING FOR STOCHASTIC NAVIER–STOKES SIGNAL OBSERVED WITH LINEAR ADDITIVE NOISE\*

FRANCESC PONS LLOPIS<sup>†</sup>, NIKOLAS KANTAS<sup>†</sup>, ALEXANDROS BESKOS<sup>‡</sup>,  
AND AJAY JASRA<sup>§</sup>

**Abstract.** We consider a nonlinear filtering problem whereby the signal obeys the stochastic Navier–Stokes equations and is observed through a linear mapping with additive noise. The setup is relevant to data assimilation for numerical weather prediction and climate modeling, where similar models are used for unknown ocean or wind velocities. We present a particle filtering methodology that uses likelihood-informed importance proposals, adaptive tempering, and a small number of appropriate Markov chain Monte Carlo steps. We provide a detailed design for each of these steps and show in our numerical examples that they are all crucial in terms of achieving good performance and efficiency.

**Key words.** stochastic Navier–Stokes, stochastic filtering, particle filters, preconditioned Crank–Nicolson Markov chain Monte Carlo

**AMS subject classifications.** 60G35, 93E11, 35Q30

**DOI.** 10.1137/17M1151900

**1. Introduction.** We focus on a stochastic filtering problem where a space-time-varying hidden signal is observed at discrete times with noise. The nonlinear filtering problem consists of computing the conditional probability law of the hidden stochastic process (the so-called signal) given observations of it collected in a sequential manner. In particular, we model the signal with a particular dissipative stochastic partial differential equation (SPDE), which is the stochastic Navier–Stokes equation (NSE). This model, or a variant thereof, is often used in applications to model unknown quantities such as atmosphere or ocean velocity. In the spirit of data assimilation and uncertainty quantification, we wish to extract information for the trajectory of the hidden signal from noisy observations using a Bayesian approach. Typical applications include numerical weather forecasting in meteorology, oceanography and atmospheric sciences, geophysics, hydrology, and petroleum engineering; see [2, 36, 6] for an overview.

We restrict to the setting where the state of interest is the time-varying velocity field,  $V(x, t)$ , in some two-dimensional bounded set  $\Omega$ . The unknown state is modeled using the stochastic NSE

$$(1) \quad dV(x, t) - \nu \Delta V(x, t) dt + B(V, V)(x, t) dt = f(x, t) dt + Q^{\frac{1}{2}} dW(x, t),$$

where  $\Delta$  is the Laplacian;  $\nu$  a viscosity constant;  $B$  a nonlinear operator due to

\*Submitted to the journal’s Methods and Algorithms for Scientific Computing section October 12, 2017; accepted for publication (in revised form) March 27, 2018; published electronically May 31, 2018.

<http://www.siam.org/journals/sisc/40-3/M115190.html>

**Funding:** The first author’s work was supported by EPSRC and the CDT in the Mathematics of Planet Earth under grant EP/L016613/1. The third author’s work was supported by the Leverhulme Trust Prize. The fourth author’s work was supported by an AcRF tier 2 grant: R-155-000-161-112 and additionally KAUST CRG4 award: 2584.

<sup>†</sup>Department of Mathematics, Imperial College London, UK (francesc.pons-llopis13@imperial.ac.uk, n.kantas@imperial.ac.uk).

<sup>‡</sup>Department of Statistical Science, University College London, UK (a.beskos@ucl.ac.uk).

<sup>§</sup>Department of Statistics and Applied Probability, National University of Singapore, Singapore (staja@nus.edu.sg).

convection;  $Q$  a positive, self-adjoint, trace class operator;  $f$  a deterministic forcing; and  $W(x, t)$  a space-time white noise as in [13]. This might appear as a restrictive choice for the dynamics, but the subsequent methodology is generic and could be potentially applied to other similar dissipative SPDEs, such as the stochastic Burger or Kuramoto–Sivashinski equations [26, 9].

The evolution of the unknown state of the SPDE is observed at discrete times and generates a sequence of noisy observations  $\mathcal{Y}_n = (Y_{t_1}, \dots, Y_{t_n})$ . In order to perform accurate estimation and uncertainty quantification, we are interested not just in approximating a single trajectory estimate of the hidden state but also in the complete filtering distribution,

$$(2) \quad \pi_n(\bullet) = \mathbb{P}[V(\cdot, t_n) \in \bullet | \mathcal{Y}_n],$$

that is, the conditional distribution of the state given all the observations obtained up to current time  $t_n$ . The main objective is to compute the filtering distribution as it evolves with time, which is an instance of the stochastic filtering problem [1]. The solution of the problem can be formulated rigorously as a recursive Bayesian inference problem posed on an appropriate function space [33]. In contrast to standard filtering problems, the problem setup here is particularly challenging: The prior consists of a complicated probability law generated by the SPDE [13], and observation likelihoods on the high-dimensional space of the signal tend to be very informative.

The aim of this paper is to propose sequential Monte Carlo (SMC) methods (also known as particle filters (PF)) that can approximate effectively these conditional distributions. Computing the evolution of the filtering distribution  $\pi_n$  is not analytically tractable, except in linear Gaussian settings. SMC is a generic Monte Carlo method that approximates the sequence of  $\pi_n$ 's and their normalizing constant  $\mathbb{P}[\mathcal{Y}_n]$  (known in statistics as marginal likelihood or evidence). This is achieved by obtaining samples known as particles and combining Importance Sampling (IS), resampling, and parallel Markov chain Monte Carlo (MCMC) steps. The main advantages of the methodology are (i) it is sequential and online in nature; (ii) it does not require restrictive model assumptions such as Gaussian noise or linear dynamics and observations; (iii) it is parallelizable, so one could gain significant speedup using appropriate hardware (e.g., GPUs, computing clusters) [35]; and (iv) it is a well-studied principled method with an extensive literature justifying its validity and theoretical properties; see, e.g., [15, 14]. So far, SMC has been extremely successful in typically low to moderate dimensions [18], but its application in high-dimensional settings has been very challenging mainly due to the difficulty to perform IS efficiently in high dimensions [44]. Despite this challenge, a few successful high-dimensional SMC implementations have appeared recently for applications with discrete time signal dynamics [39, 49, 48, 50, 6, 10, 3].

We will formulate the filtering problem with discrete time observations and continuous time dynamics. This setup has appeared previously in [43, 42] for signals corresponding to low-dimensional stochastic differential equations (SDEs). The aim of this paper is to provide a novel, accurate, and more efficient SMC design when the hidden signal is modeled by an SPDE with linear Gaussian observation. To achieve this challenging task, the particle filter will use computational tools that have been previously successful in similar high-dimensional problems, such as tempering [31] and preconditioned Crank–Nicholson MCMC steps [25, 11]. Using such tools, we propose a particle algorithm that can be used to approximate  $\pi_n$  when the signal obeys the stochastic NSE and the observations are linear with additive noise. On a general level, the proposed algorithm has a similar structure to [29], but here we additionally adopt

the use of IS. We will provide a detailed design of the necessary likelihood informed importance proposals and the MCMC moves used. We extend known IS techniques for SDEs [24, 51] and MCMC moves for high-dimensional problems [46, 25, 11] to make them applicable for filtering problems involving the stochastic NSE or other dissipative SPDEs. In the context of particle filtering, our developments lead to an SMC algorithm that performs effectively for the high-dimensional problem at hand using a moderate amount of particles.

The material presented in this paper can be viewed as an extension of some ideas in the authors' earlier work in [31]. In [31] we considered the deterministic NSE with more general observation equations. In the present paper the model for the signal contains additive noise, and we assume linear observation schemes. This allows for the possibility of using likelihood-informed importance proposals, and the MCMC steps need to be designed to be invariant to a more complicated conditional law due to the SPDE dynamics. The organization of this paper is as follows. In section 2 we present some background on the stochastic NSE, and in section 3 we formulate the filtering problem of interest. In section 4 we present the SMC algorithm, and in section 5 we present a numerical case study that illustrates the performance and efficiency of our method. Finally, in section 6 we provide some concluding remarks.

**2. Background on the stochastic NSE.** We present some background on the two-dimensional stochastic NSE defined on an appropriate separable Hilbert space. We restrict the presentation to the case of periodic boundary conditions following the treatment in [20]. This choice is motivated mainly for convenience in exposition and for performing numerical approximations using fast Fourier transforms (FFT). The formulation and properties of the stochastic NSE can allow for the more technically demanding Dirichlet conditions on a smooth boundary [21, 25]. We stress that the subsequent particle filtering methodology is generic and does not rely on the choice of boundary conditions.

**2.1. Preliminaries.** Let the region of interest be the torus  $\Omega := [0, 2\pi]^2$  with  $x = (x_1, x_2) \in \Omega$  being a point on the space. The quantity of interest is a time-space-varying velocity field  $v : \Omega \times [0, T] \rightarrow \mathbb{R}^2$ ,  $v(x, t) = (v_1(x, t), v_2(x, t))'$  and  $v(\cdot, 0, t) = v(\cdot, 2\pi, t)$  due to the periodic boundary conditions; here  $\cdot$  denotes vector/matrix transpose. It is convenient to work with the Fourier characterization of the function space of interest,

$$(3) \quad H = \left\{ u = \sum_{k \in \mathbb{Z}^2 \setminus \{0\}} u_k \psi_k(x) \mid u_{-k} = -\overline{u_k}, \sum_{k \in \mathbb{Z}^2 \setminus \{0\}} |u_k|^2 < \infty \right\},$$

using the following orthonormal basis functions for  $H$ :

$$\psi_k(x) = \frac{1}{2\pi} \frac{k^\perp}{|k|} e^{i k \cdot x}, \quad k \in \mathbb{Z}^2 \setminus \{0\}, \quad k^\perp := (-k_2, k_1)'.$$

The deterministic NSE is given by the following the functional evolution equation:

$$(4) \quad dv + \nu A v dt + B(v, v) dt = f(t) dt, \quad v(0) \in H,$$

Following standard notation, we denote  $P : (L^2_{\text{per}}(\Omega))^2 \rightarrow H$  for the Leray projector ( $L^2_{\text{per}}(\Omega)$  is the space of squared-integrable periodic functions),  $A := P(-\Delta) = -\Delta$  for the Stokes operator,  $B(u, v) = P((u \cdot \nabla)v)$  for the convection mapping, and  $f \in L^2(0, T; H)$  for the forcing.

One can introduce additive noise in the dynamics in a standard manner. First, we define the upper half-plane of wavenumbers

$$\mathbb{Z}_+^2 = \{k = (k_1, k_2) \in \mathbb{Z}^2 \setminus \{0\} : k_1 + k_2 > 0\} \\ \cup \{k = (k_1, k_2) \in \mathbb{Z}^2 \setminus \{0\} : k_1 + k_2 = 0, k_1 > 0\}.$$

Let

$$Z_k(t) = Z_k^{re}(t) + i Z_k^{im}(t), \quad k \in \mathbb{Z}_+^2,$$

where  $\{Z_k^{re}, Z_k^{im}\}$  are (independent) standard Brownian motions on  $[0, T]$ . In the spirit of [13, section 4.1], consider a covariance operator  $Q$  such that  $Q\psi_k = \sigma_k^2 \psi_k$  for  $\sigma_k^2 > 0$ ,  $\sigma_{-k} = \sigma_k$ . Then we can define the  $Q$ -Wiener process as

$$(5) \quad Q^{\frac{1}{2}}W(t) := \sum_{k \in \mathbb{Z}^2 \setminus \{0\}} \sigma_k Z_k(t) \psi_k(x)$$

under the requirement  $Z_{-k} \equiv -\overline{Z_k}$ ,  $k \in \mathbb{Z}_+^2$ . Thus, we are working with a diagonal covariance matrix (w.r.t. the relevant basis of interest), though other choices could easily be considered. We will also work under the scenario that  $\sigma_k^2 = O(|k|^{-2(1+\epsilon)})$  for some  $\epsilon > 0$ , so that  $\sum_{k \in \mathbb{Z}^2 \setminus \{0\}} \sigma_k^2 < \infty$ , i.e.,  $Q$ , is trace-class operator. Finally, we will use  $\mathbb{W}(\cdot)$  to denote the  $Q$ -Wiener measure on  $[0, T]$ .

Having introduced the random component, we are now interested in weak solutions  $V = (V(t))_{t \in [0, T]}$  of the functional SDE,

$$(6) \quad dV(t) + \nu AV(t) dt + B(V(t), V(t))dt = f(t) dt + Q^{\frac{1}{2}}dW(t), \quad V(0) = v_0,$$

with the solution understood pathwise on the probability space  $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t \geq 0}, \mathbb{P})$ . More formally, following [20], we define the spaces

$$\mathcal{V}_s := \left\{ u = \sum_{k \in \mathbb{Z}^2 \setminus \{0\}} u_k \psi_k(x) \mid u_{-k} = -\overline{u_k}, \sum_{k \in \mathbb{Z}^2 \setminus \{0\}} |k|^{2s} |u_k|^2 < \infty \right\}, \quad s \in \mathbb{R}.$$

Since the operator  $Q^{1/2}$  is linear and bounded in  $H$  and  $\text{Im}(Q^{1/2}) \equiv \mathcal{V}_{1+\epsilon}$  [20, Theorem 6.1] implies that for  $v_0 \in \mathcal{V}_1$  and  $f \in C([0, T]; \mathcal{V}_1)$ , there exists a unique solution for (6) such that  $V \in C([0, T]; \mathcal{V}_1)$ . In [20, 21] one may also find more details on the existence of an invariant distribution, together with irreducibility and Feller properties of the corresponding Markov transition kernel.

**2.2. Galerkin projections and computational considerations.** Using the Fourier basis (3), we can write the solution as

$$V(t) = \sum_{k \in \mathbb{Z}^2 \setminus \{0\}} u_k(t) \psi_k(x), \quad u_{-k}(t) \equiv -\overline{u_k(t)}, \\ u_k(t) = \langle V(t), \psi_k \rangle = \int_{\Omega} V(t) \cdot \overline{\psi_k(x)} dx.$$

Hence, it is equivalent to consider the parameterization of  $V$  via  $\{u_k(t)\}_{k \in \mathbb{Z}^2 \setminus \{0\}}$ . By taking the inner product with  $\psi_k$  on both sides of (6), it is straightforward to obtain that the  $u_k$ 's obey the infinite-dimensional SDE

$$(7) \quad du_k(t) = -\nu |k|^2 u_k(t) dt \\ - \sum_{m, p \in \mathbb{Z}^2 \setminus \{0\}} b_{k, m, p} u_m(t) u_p(t) dt + f_k(t) dt + \sigma_k dZ_k(t), \quad k \in \mathbb{Z}_+^2$$

with

$$b_{k,m,p} = \langle B(\psi_m, \psi_p), \psi_k \rangle, \quad f_k(t) = \langle f(t), \psi_k \rangle.$$

Recall that due to  $V(t)$  being a real field,  $u_{-k}(t) \equiv -\overline{u_k(t)}$ ,  $k \in \mathbb{Z}_+^2$ . This parameterization of  $V$  is more convenient, as it allows performing inference on a vector (even if infinitely long), with coordinates evolving according to an SDE. For numerical purposes one is forced to use Galerkin discretisations, using projections of  $V$  onto a finite Hilbert space instead. Consider the set of wavenumbers in

$$\mathbb{L} = \{k \in \mathbb{Z}_+^2 : (k_1 \vee k_2) \leq L\}$$

for some integer  $L > 0$ , and define the finite-dimensional subspace  $H_L$  via the projection  $P_L : H \rightarrow H_L$  so that

$$P_L v = \sum_{k \in \mathbb{L}} \langle v, \psi_k \rangle \psi_k.$$

Then, inferring the Galerkin projection for  $V$  corresponds to inferring the vector  $\{u_k(t)\}_{k \in \mathbb{L}}$  that obeys the following finite-dimensional SDE:

$$(8) \quad du_k(t) = -\nu |k|^2 u_k(t) - \sum_{m,p \in \mathbb{L}} b_{k,m,p} u_m(t) u_p(t) dt + f_k(t) dt + \sigma_k dZ_k(t), \quad k \in \mathbb{L}.$$

This high-dimensional SDE will provide an approximation for the infinite-dimensional SPDE. Such an inference problem is more standard but is still challenging due to the high dimensionality of  $\mathbb{L}$  and the nonlinearities involved in the summation term of the drift function in (8). Since (8) is only an approximation of (7), it will induce a bias in the inferential procedure. In our paper, we do not study the size of this bias. Instead, we concentrate our efforts on designing an algorithm to approximate  $\pi_n$  (in (2)) that is robust to *mesh refinement*. This means our method should perform well numerically when one increases  $L$  (and, indeed, reducing the bias in the numerical approximation of (7)). Naturally this would be at the expense of adding computational effort at a moderate amount, but this will depend on the particular numerical scheme used to approximate the solution of (8). For instance, for the FFT-based numerical schemes used in section 5, the computational cost is  $\mathcal{O}(L^2 \log L)$ .

**2.3. The distribution of  $v_0$ .** We assume that the initial condition of  $V$  is random and distributed according to the Gaussian process prior

$$(9) \quad \pi_0 = \mathcal{N}(\mu, \beta^2 A^{-\alpha}), \quad \alpha > 2, \beta > 0, \mu \in \mathcal{V}_1$$

with hyperparameters  $\alpha, \beta$  affecting the roughness and magnitude of the initial vector field. This is a convenient but still flexible enough choice of a prior; see [13, sections 2.3 and 4.1] for more details on Gaussian distributions on Hilbert spaces. Notice that  $\pi_0$  admits the Karhunen–Loève expansion

$$\pi_0 = \mathcal{L}aw \left( \sum_{k \in \mathbb{Z}^2 \setminus \{0\}} \left( \mu_k + \frac{\beta}{\sqrt{2}} |k|^{-\alpha} \xi_k \psi_k \right) \right)$$

with  $\mu_k = \langle \mu, \psi_k \rangle$ ,  $k \in \mathbb{Z}_+^2$  (so, necessarily,  $\mu_{-k} = -\overline{\mu_k}$ ,  $k \in \mathbb{Z}_+^2$ ) and

$$\operatorname{Re}(\xi_k), \operatorname{Im}(\xi_k) \stackrel{iid}{\sim} \mathcal{N}(0, 1), \quad k \in \mathbb{Z}_+^2; \quad \xi_{-k} = -\overline{\xi_k}, \quad k \in \mathbb{Z}_+^2.$$

Since the covariance operator is determined via the Stokes operator  $A$ , one can easily check that the choice  $\alpha > 2$  implies that for  $v_0 \in \mathcal{V}_1$ ,  $\pi_0$ -a.s.; thus, the conditions for existence of weak solution of (6) in [20, Theorem 6.1] are satisfied a.s. in the initial condition. Notice that sampling from  $\pi_0$  is straightforward.

**3. The stochastic filtering problem.** In section 2 we defined the SPDE providing the unknown signal, i.e., the object we are interested in performing Bayesian inference upon. In this section we present the nonlinear filtering problem in detail. We begin by discussing the observations. We assume that the vector field  $V$  is unknown but generates a sequence of noisy observations  $\mathcal{Y}_n = (Y_{t_1}, \dots, Y_{t_n})$  at ordered discrete time instances  $(t_p)_{p=1, \dots, n}$  with  $t_n < t_{n+1} < T$  for all  $n$ , with  $Y_{t_i} \in \mathbb{R}^{d_y}$ , for  $d_y \geq 1$ . Each observation vector  $Y_{t_i}$  is further assumed to originate from the following observation equation:

$$(10) \quad Y_{t_n} = FV(t_n) + \Xi_n, \quad \Xi_n \sim \mathcal{N}(0, \Sigma),$$

where  $F$  is a bounded linear operator  $F : H \rightarrow \mathbb{R}^{d_y}$  and  $\Sigma \in \mathbb{R}^{d_y \times d_y}$  is symmetric positive-definite. One can then write the observation likelihood at instance  $t_n$  as

$$p(Y_{t_n} | V(t_n)) = \frac{\exp\left(-\frac{1}{2} \left| \Sigma^{-\frac{1}{2}} (Y_{t_n} - FV(t_n)) \right|^2\right)}{(2\pi)^{d_y/2} |\Sigma|^{1/2}}.$$

Using a linear observation model is restrictive, but it does include typical observation schemes used in practice. We focus our attention on the case when  $Y_{t_n}$  is a noisy measurement of the velocity field at different fixed stationary points  $x_l \in \Omega$ ,  $l = 1, \dots, p$ . This setting is often referred to as *Eulerian data assimilation*. In particular, we have that

$$F = (F'_1, \dots, F'_p)'$$

with  $F_l$  denoting a spatial average over a (typically small) region around  $x_l$ ,  $l = 1, \dots, p$ , say  $B_{x_l}(r) = \{x \in \Omega : |x - x_l| \leq r\}$ , for some radius  $r > 0$ ; that is,  $F_l$  is the integral operator

$$(11) \quad F_l V(t) = \frac{1}{|B_{x_l}(r)|} \int_{B_{x_l}(r)} V(t, x) dx$$

with  $|B_{x_l}(r)|$  denoting the area of  $B_{x_l}(r)$ . In what follows, other integral operators could also be similarly used, such as  $F_l V(t) = (\int_{\Omega} V(t, x) w_{x_l}(x) dx) / (\int_{\Omega} w_{x_l}(x) dx)$ , with  $w_{x_l} \in L^2(\Omega)$  being appropriate weighting functions that decay as  $|x - x_l|$  grows.

Earlier in the introduction, the filtering problem was defined as the task of computing the conditional distribution  $\pi_n(\cdot) = \mathbb{P}[V(t_n) \in \cdot | \mathcal{Y}_n]$ . Due to the nature of the observations, it is clear we are dealing with a discrete time filtering problem. A particular challenge here (in common with other typical nonlinear SPDEs) is that the distribution of the associated Markov transition kernel,  $\mathbb{P}[V(t_n) \in \cdot | V(t_{n-1}) = v]$ , is intractable. Still, it is possible to simulate from the unconditional dynamics of  $V(t)$  given  $V(t_{n-1}) = v$  using standard time discretization techniques. (The simulated path introduces a time discretization bias, but its effect is ignored in this paper.)

We aim to infer the following posterior distribution based on the *continuous time signal*

$$\Pi_n(\cdot) = \mathbb{P}[V^n \in \cdot | \mathcal{Y}_n], \quad V^n := (V(t))_{t \in [0, t_n]};$$

we also denote

$$V_{n-1}^n = (V(t))_{t \in (t_{n-1}, t_n]}.$$

This data augmentation approach—when applying importance sampling on continuous time—has appeared in [25] for a related problem and in [43] for filtering problems involving certain multivariate SDEs. We proceed by writing the filtering recursion for  $\Pi_n$ . We denote the law of  $V$  in (6) for the time interval between  $t_{n-1}$  and  $t_n$  as

$$\mathbb{V}_{n-1}^n(\cdot|v) := \mathbb{P}[(V(t))_{t \in (t_{n-1}, t_n]} \in \cdot | V(t_{n-1}) = v].$$

Then one may use Bayes's rule to write  $\Pi_n$  recursively as

$$(12) \quad \frac{d\Pi_n}{d(\Pi_{n-1} \otimes \mathbb{V}_{n-1}^n)}(V^n) = \frac{p(Y_{t_n}|V(t_n))}{p(Y_{t_n}|\mathcal{Y}_{n-1})},$$

where  $p(Y_{t_n}|\mathcal{Y}_{n-1}) = \int p(Y_{t_n}|V(t_n))[\Pi_{n-1} \otimes \mathbb{V}_{n-1}^n](dV^n)$ .

In addition, one can attempt to propose paths from an appropriate SPDE different from (6), say

$$(13) \quad \begin{aligned} d\tilde{V}(t) + \nu A\tilde{V}(t)dt + B(\tilde{V}(t), \tilde{V}(t))dt \\ = Q^{\frac{1}{2}}g(t, \tilde{V}(t))dt + f(t)dt + Q^{\frac{1}{2}}dW(t), \quad t \in (t_{n-1}, t_n], \end{aligned}$$

where  $g : [0, T] \times H \mapsto H$  and  $Q^{\frac{1}{2}}W_t$  is a  $Q$ -Wiener process on  $(t_{n-1}, t_n]$ . We define

$$\mathbb{Q}_{n-1}^n(\cdot|v) := \mathbb{P}[\tilde{V}_{n-1}^n \in \cdot | \tilde{V}(t_{n-1}) = v].$$

One needs to ensure that the change of drift  $g$  is appropriately chosen so that a Girsanov theorem holds and  $\mathbb{V}_{n-1}^n(\cdot|v)$  is absolutely continuous with respect to  $\mathbb{Q}_{n-1}^n(\cdot|v)$  for all relevant  $v$  with the recursion in (12) becoming

$$(14) \quad \frac{d\Pi_n}{d(\Pi_{n-1} \otimes \mathbb{Q}_{n-1}^n)}(V^{n-1}, \tilde{V}_{n-1}^n) \propto p(Y_{t_n}|\tilde{V}(t_n)) \cdot \frac{d\mathbb{V}_{n-1}^n}{d\mathbb{Q}_{n-1}^n}(\tilde{V}_{n-1}^n|V(t_{n-1})).$$

Here  $(V^{n-1}, \tilde{V}_{n-1}^n)$  are assumed to be typical elements of the sample space of either of the two probability measures above (e.g., all such paths are assumed to possess relevant continuity properties at  $t_{n-1}$ ).

In the context of particle filtering and IS, one aims to design  $g$  in a way that the proposed trajectories are in locations where  $\Pi_n$  is higher. This in turn implies that the importance weights in (14) will exhibit much less variance than the ones from the prior signal dynamics; hence, the design of  $g$  is critical for generating effective Monte Carlo approximations.

**4. Particle filtering.** We are interested in approximating the distribution  $\Pi_n$  using a particle filter approach. We present in Algorithm 1 a naive particle filter algorithm that provides the particle approximations:

$$\Pi_n^N = \sum_{j=1}^N \mathcal{W}_n^j \delta_{V^j} \quad \text{or} \quad \bar{\Pi}_n^N = \frac{1}{N} \sum_{j=1}^N \delta_{V^j}.$$

Such a particle filter will be typically overwhelmed by the dimensionality of the problem and will not be able to provide accurate solutions with a moderate computational cost. When  $g = 0$  in 13, the algorithm corresponds to a standard bootstrap particle filter. For the latter, it is well known in the literature [6, 44] that it exhibits weight degeneracy in the presence of large dissimilarity between  $\Pi_{n-1} \otimes \mathbb{V}_{n-1}^n$  and  $\Pi_n$ , which

---

**Algorithm 1.** A Naive Particle Filter.

---

- Initialize  $V_0^i \sim \pi_0$ ,  $1 \leq i \leq N$ .
- For  $n \geq 1$ 
  1. For  $i = 1, \dots, N$ : sample independently:

$$\tilde{V}_{n-1}^{n,i} \sim \mathbb{Q}_{n-1}^n(\cdot | V(t_{n-1})^i).$$

2. For  $i = 1, \dots, N$ : compute importance weights:

$$\mathcal{W}_n^i \propto p(Y_{t_n} | \tilde{V}(t_n)^i) \cdot \frac{d\mathbb{V}_{n-1}^n}{d\mathbb{Q}_{n-1}^n}(V_{n-1}^{n,i} | V(t_{n-1})^i), \quad \text{s.t.} \quad \sum_{i=1}^N \mathcal{W}_n^i = 1.$$

3. For  $i = 1, \dots, N$ : resample:

$$V^{n,i} \sim \sum_{j=1}^N \mathcal{W}_n^j \delta_{(V^{n-1,j}, \tilde{V}_{n-1}^{n,j})}(\cdot).$$


---

can be caused in our context by the high dimensionality of the state space and the complexity of the SPDE dynamics. When  $g$  is well designed, the particles can be guided in areas of larger-importance weights, and the algorithmic performance can be considerably improved, but this modification may still not be sufficient for obtaining a robust and efficient algorithm.

In the remainder of this section, we will discuss how to improve upon this first attempt to tackle the high-dimensional filtering problem at hand using the following ingredients: (i) specifying a particular form of  $g$  in (13) that results in gains of efficiency, (ii) using adaptive tempering, and (iii) MCMC moves. Guided proposals and tempering are employed to bridge the dissimilarity between  $\Pi_{n-1} \otimes \mathbb{V}_{n-1}^n$  and  $\Pi_n$ . The MCMC steps are required for injecting additional diversity in the particle population, which would otherwise diminish gradually due to successive resampling and tempering steps. The method is summarized in Algorithm 2. In the following subsections, we explain in detail our implementation of (i)–(iii) mentioned above.

**4.1. Likelihood-informed proposals.** In the importance weight of (14) we are using a Girsanov theorem and assume absolute continuity between SPDEs (13) and (6) when started at the same position. Under the assumption

$$(15) \quad \mathbb{P} \left[ \int_0^T \|g(t, V(t))\|^2 dt < \infty \right] = 1,$$

absolute continuity indeed holds, and we have Radon–Nikodym derivative

$$\begin{aligned} \log \frac{d\mathbb{V}_{n-1}^n}{d\mathbb{Q}_{n-1}^n}(\tilde{V}_{n-1}^n | V^{n-1}(t_{n-1})) \\ = - \int_{t_{n-1}}^{t_n} \langle Q^{\frac{1}{2}} g(t, \tilde{V}(t)), Q^{\frac{1}{2}} dW(t) \rangle_0 - \frac{1}{2} \int_{t_{n-1}}^{t_n} \left\| Q^{\frac{1}{2}} g(t, \tilde{V}(t)) \right\|_0^2 dt, \end{aligned}$$

where

$$\langle u, v \rangle_0 := \langle Q^{-\frac{1}{2}} u, Q^{-\frac{1}{2}} v \rangle \equiv \sum_{k \in \mathbb{Z}^2 \setminus \{0\}} \frac{1}{\sigma_k^2} \langle u, \psi_k \rangle \langle v, \psi_k \rangle;$$



**Algorithm 2.** Adaptive Particle Filtering Algorithm.

- At  $n = 0$ . For  $i = 1, \dots, N$ , sample i.i.d.  $V_0^i \sim \pi_0$ , and set  $\mathcal{W}_0^i = 1/N$ .
- At time  $n \geq 1$ .
  1. For  $i = 1, \dots, N$ : sample independently

$$X_n^i \sim \mathbb{Q}_{n-1}^n(\cdot | V^{n-1,i}(t_{n-1}))$$

2. Set  $l = 0$ ,  $X_{n,0}^i = X_n^i$ ,  $\Pi_{n,0} = \Pi_{n-1} \otimes \mathbb{Q}_{n-1}^n$ ,  $\phi_{n,0} = 0$ .
3. While  $\phi_{n,l} < 1$ 
  - (a) Set  $l \leftarrow l + 1$
  - (b) Specify  $\Pi_{n,l}$ ,  $\phi_{n,l}$  based on the ESS computation in (19)
  - (c) For  $i = 1, \dots, N$ 
    - i. Compute weights  $\mathcal{W}_{n,l}^i$  as in (18)
    - ii. Resample and move particles:

$$X_{n,l}^i \stackrel{i.i.d.}{\sim} \sum_{j=1}^N \frac{\mathcal{W}_{n,l}^j}{\sum_{k=1}^N \mathcal{W}_{n,l}^k} \mathcal{K}_{n,l}^m(\cdot | X_{n,l-1}^j)$$

4. If  $\phi_{n,l} = 1$  return  $V^{n,i} = (V^{n-1,i}, X_{n,l}^i)$ ,  $\tau_n = l$ ; otherwise go back to Step 3.

see [13, Theorem 10.14] and [13, Lemma 10.15] for details. It remains to provide an effective design for  $g$ . One can use proposals developed for problems whereby a finite-dimensional SDE generates linear Gaussian observations and one is interested in performing a similar IS method; see, e.g., [24, 51, 40, 41, 47]. In this paper we use the proposal employed in [24] and set

$$(16) \quad g(t, V(t)) = Q^{\frac{1}{2}} F^* (\Sigma + (t_n - t) F Q F^*)^{-1} (Y_{t_n} - F V(t)), \quad t \in (t_{n-1}, t_n],$$

where  $F^*$  denotes the adjoint of  $F$ . The guiding function  $g$  could be interpreted as a one-step Euler approximation of the  $h$ -transform needed to evolve  $V(t)$  conditional on the acquired observation  $Y_{t_n}$  within the interval  $(t_{n-1}, t_n]$ . It is not hard to verify (15) for this choice of  $g$ . Since  $\Sigma, Q$  are invertible,  $(\Sigma + (t_n - t) F Q F^*)^{-1}$  exists via the Sherman–Morrison–Woodbury identity and  $Q^{\frac{1}{2}} F^* (\Sigma + (t_n - t) F Q F^*)^{-1}$  is a bounded linear operator. Then (15) holds from [13, Proposition 10.18] and [32, Proposition 2.4.9], which imply that there exists a  $\delta > 0$  such that

$$\sup_{t \in [0, T]} \mathbb{E} \left[ \exp \left( \delta \|g(t, V(t))\|^2 \right) \right] < \infty,$$

which implies (15).

For the finite-dimensional SDE case, more elaborate guiding functions can be found in [51, 47], and some of these could be potentially extended so that they can be used in the SPDE setting instead of (16). The advantage of using  $g$  in (16) is that it provides a simple functional and can perform well for problems where  $t_n - t_{n-1}$  is of moderate length, as also confirmed in the numerical examples of section 5.

**4.2. Bridging  $\Pi_{n-1}$  and  $\Pi_n$  with adaptive tempering.** Guided proposals aim to bridge the dissimilarity between  $\Pi_{n-1} \otimes \mathbb{V}_{n-1}^n$  and  $\Pi_n$  by considering a Bayesian update from  $\Pi_{n-1} \otimes \mathbb{Q}_{n-1}^n$  to  $\Pi_n$ . In a high-dimensional setting, even using well-designed likelihood-informed proposals is not sufficient to bridge the dissimilarity

between the informed proposal  $\Pi_{n-1} \otimes \mathbb{Q}_{n-1}^n$  and the target  $\Pi_n$ . As a result, the importance weights could still degenerate. To avoid this, more effort is required. One possibility is to allow for a progressive update via a sequence of intermediate artificial distributions between  $\Pi_{n-1} \otimes \mathbb{Q}_{n-1}^n$  to  $\Pi_n$ , which we will denote as  $\Pi_{n,l}$  with  $l = 1, \dots, \tau_n$  and require that  $\Pi_{n,0} = \Pi_{n-1} \otimes \mathbb{Q}_{n-1}^n$  and  $\Pi_{n,\tau_n} = \Pi_n$ . This is a well-known strategy to improve particle filters; see [38, 23] for some early works in this direction for low-dimensional problems.

To construct the sequence, we will use a standard tempering scheme [37]. Each  $\Pi_{n,l}$  can be defined using

$$(17) \quad \frac{d\Pi_{n,l}}{d(\Pi_{n-1} \otimes \mathbb{Q}_{n-1}^n)}(V^{n-1}, \tilde{V}_{n-1}^n) \propto \left( \frac{dV_{n-1}^n}{d\mathbb{Q}_{n-1}^n}(\tilde{V}_{n-1}^n) \cdot p(Y_{t_n} | \tilde{V}(t_n)) \right)^{\phi_{n,l}}$$

for inverse temperatures  $0 = \phi_{n,0} < \phi_{n,1} < \dots < \phi_{n,\tau_n} = 1$ . Note that each  $\Pi_{n,l}$  is defined on the same state space of  $V^n$  and there are no natural stochastic dynamics connecting each  $\Pi_{n,l}$ . As a result, we will follow the framework of [16, 8] and use artificial dynamics provided by a MCMC transition kernel that is invariant to  $\Pi_{n,l}$ . The details are provided in the next section. Using these MCMC proposals will result in the weights at iteration  $(n, l)$  being  $\mathcal{W}_{n,l}^j \propto \frac{d\Pi_{n,l}}{d\Pi_{n,l-1}}$ , which depends on  $\phi_{n,l} - \phi_{n,l-1}$  and the proposed  $V^n$  from the MCMC kernel.

The main issue that needs addressing for this scheme to be successful is how to determine the temperatures  $\phi_{n,l}$  and their number  $\tau_n$ . We propose to set these on-the-fly using an adaptive procedure introduced in [27]. Assume we are at the  $n$ th step of the algorithm, have completed  $l-1$  tempering steps, and have equally weighted particles. The next temperature is determined by expressing the weights as a function of  $\phi$ ,

$$(18) \quad \mathcal{W}_{n,l}^j(\phi) \propto \left( \frac{dV_{n-1}^n}{d\mathbb{Q}_{n-1}^n}(\tilde{V}_{n-1}^{n,j}) p(Y_{t_n} | \tilde{V}(t_n)^j) \right)^{\phi - \phi_{n,l-1}}, \quad \phi_{n,l-1} < \phi \leq 1,$$

$$\sum_{i=1}^N \mathcal{W}_{n,l}^i(\phi) = 1,$$

and determining  $\phi_{n,l}$  via a requirement based on a quality criterion for the particle population. We use here the effective sample size (ESS) and set

$$(19) \quad \phi_{n,l} = \inf \left\{ \phi \in (\phi_{n,l-1}, 1] : ESS_{n,l}(\phi) := \frac{1}{\sum_{j=1}^N \{\mathcal{W}_{n,l}^j(\phi)\}^2} \leq \alpha N \right\}$$

(under the convention that  $\inf \emptyset = 1$ ) with a user-specified fraction  $\alpha \in (0, 1)$ . Equation (19) can be easily solved numerically using, for instance, a standard bisection method. This approach leads to a particle approximation for  $\Pi_{n,l}$ , say

$$\Pi_{n,l}^N = \sum_{i=1}^N \mathcal{W}_{n,l}^i(\phi_{n,l}) \delta_{V^{n,i}};$$

we then propose to resample from  $\Pi_{n,l}^N$  so that one ends up with equally weighted particles.

The adaptive tempering procedure is presented in step 3 of Algorithm 2. In steps 3(a)–3(c), (18)–(19) are followed by resampling and MCMC steps and the steps are iterated until  $\phi_{n,l} = 1$ . The MCMC dynamics are denoted by  $\mathcal{K}_{n,l}^m$  and will be

discussed below. For every  $n$ , the output of step 4 of Algorithm 2 provides a particle approximation  $\Pi_n^N = \frac{1}{N} \sum_{i=1}^N \delta_{V_{n,i}}$  targeting  $\Pi_n$ . The interesting feature of this algorithm is that when moving from  $\Pi_{n-1}$  to  $\Pi_n$ , it does not require a user-specified intermediate sequence of target distributions  $(\Pi_{n,l})_{l=0,\dots,\tau_n}$ , but these are adaptively set according to the locations of the particles and (19). The number of steps required,  $\tau_n$ , will be determined according to the difficulty in assimilating  $Y_{t_n}$ .

*Remark 4.1.* The convergence of Algorithm 2 has been studied in [4, 22].

*Remark 4.2.* In Algorithm 2, for simplicity we always resample once  $\phi_{n,\tau_n} = 1$ . This can be avoided, but then in the next time-step of the algorithm one should use

$$\mathcal{W}_{n+1,0}^j(\phi) = \mathcal{W}_{n,\tau_n}^j \cdot \left( \frac{dV_n^{n+1}}{dQ_n^{n+1}}(\tilde{V}_n^{n+1,j}) \cdot p(Y_{t_{n+1}} | \tilde{V}(t_{n+1})^j) \right)^\phi.$$

**4.3. Adding particle diversity with MCMC kernels.** Successive resampling due to the tempering steps leads to sample impoverishment unless the method reinjects sampling diversity. To achieve this, we propose using a small number of iterations from a MCMC procedure that leaves  $\Pi_{n,l}$  invariant. This is not the only possible choice, but it does lead to a simple weight expression seen above; see [16] for extensions and more details. We use a particular MCMC design similar to [25] that is well defined on function spaces (based on theory for MCMC on general state spaces [46]). The design is often referred to as preconditioned Crank–Nicolson, abbreviated here to pCN; see [45, 11] for a detailed review.

We begin with a basic description of the pCN scheme for a given target distribution  $\Pi$ ; for simplicity, we will drop the subscripts  $n, l$  here. We will denote the one-step Markov probability kernel obtained from the MCMC procedure as

$$(20) \quad \mathcal{K}[V' \in \cdot | V] = \alpha(V, V') \mathcal{Q}[V' \in \cdot | V] + \delta_V(\cdot) \left( 1 - \int \alpha(V, V') \mathcal{Q}[dV' | V] \right)$$

with  $\mathcal{Q}$  denoting the proposal kernel and  $\alpha$  the acceptance probability in a standard Metropolis–Hastings framework. Let  $\Lambda$  be a probability measure that is absolutely continuous with respect to  $\Pi$  with Radon–Nikodym derivative

$$\frac{d\Pi}{d\Lambda}(V) =: \vartheta(V).$$

Similar to [45, 11, 25] we specify the proposal kernel  $\mathcal{Q}$  to satisfy detailed balance with respect to  $\Lambda$ , i.e.,  $\mathcal{Q}(dV'|V)\Lambda(dV) = \mathcal{Q}(dV|V')\Lambda(dV')$ . Then using

$$\alpha(V, V') = 1 \wedge \frac{\vartheta(V')}{\vartheta(V)}$$

provides a kernel  $\mathcal{K}$  which is  $\Pi$ -invariant (by [46, Theorem 2]).

Next we discuss implementing the pCN design for our problem. At iteration  $(n, l)$  the target distribution for the MCMC kernels is  $\Pi_{n,l}$ , so let  $\mathcal{K}_{n,l}$ ,  $\mathcal{Q}_{n,l}$ , and  $\alpha_{n,l}$  denote the corresponding MCMC kernel, proposal, and acceptance ratio, respectively. Note that the state space of  $\Pi_{n,l}$  is the space of paths  $V^n$ , which is growing with each observation time  $n$ . We stress that for the purpose of particle filtering we are mainly interested in the invariance property of  $\mathcal{K}_{n,l}$  (to  $\Pi_{n,l}$ ) and not necessarily its ergodic properties on the full space. With this in mind  $\mathcal{Q}_{n,l}$  can be a Markov kernel that generates proposals  $V'$  with  $V'_s = V_s$  for  $s \leq t_{n-1}$ . This allows for online computation

at each  $n, l$ . At the same time reversibility holds, as Proposition 1 and Theorem 2 in [46] still hold for such proposals. From a practical perspective, we are adding noise to the path of the hidden signal only within  $(t_{n-1}, t_n]$ .

Then we need to specify  $\Lambda_n$  and  $\mathcal{Q}_{n,l}$ . Recall that for a fixed  $n$  the state space of each  $\Pi_{n,l}$  is the same for different  $l$ , so  $\Lambda_n$  needs not vary with  $l$ . One possibility is to let  $\Lambda_n = \Pi_{n-1} \otimes \mathbb{Q}_{n-1}^n$  and suppose  $V_{n-1}^n = V_{n-1}^n(W)$  with  $W$  being the driving noise that generated  $V_{n-1}^n$ . Note that we can assume that  $W(t_{n-1}) = 0$  without loss of generality since the  $V$ -path uses the increments of  $W$ . Suppose also that both  $V_{n-1}^n$  and  $W$  are stored in the computer's memory so that

$$\vartheta_{n,l}(V^{n-1}, \tilde{V}_{n-1}^n) = \frac{d\Pi_{n,l}}{d\Lambda_n}(V^{n-1}, \tilde{V}_{n-1}^n) = \left( \frac{dV_{n-1}^n}{d\mathbb{Q}_{n-1}^n}(\tilde{V}_{n-1}^n) p(Y_{t_n} | \tilde{V}^n(t_n)) \right)^{\phi_{n,l}}.$$

To simulate from a  $\Lambda_n$ -preserving proposal one first generates a new noise sample  $W'$ :

$$(21) \quad W(s)' = \rho W(s) + \sqrt{1 - \rho^2} \xi(s), \quad t_{n-1} < s \leq t_n, \quad \xi \sim \mathbb{W},$$

where  $W(s)$  is the noise driving  $V$  and  $\mathbb{W}$  is the Q-Wiener measure. To return to the original space, we use the new noise  $W'$  to solve for  $V'$  in (13). A standard calculation can show that  $W' \sim \mathbb{W}$ , which in turn implies that for the part of the proposal  $V'$  in  $(t_{n-1}, t_n]$ ,  $(V_{n-1}^n)' \sim \mathbb{Q}_{n-1}^n$  holds. Reversibility with respect to  $\Lambda$  is ensured using a simple conditioning and marginalization argument.

In Algorithm 2 we use  $m$  iterations of (20) with  $\mathcal{Q}_{n,l}$  specified as above. The corresponding  $m$ -iterate of the MCMC transition kernel is denoted as  $\mathcal{K}_{n,l}^m$  and is presented in Algorithm 3 in an algorithmic form. To simplify exposition, in Algorithm 2, for each iteration  $(n, l)$  the simulated tempered path  $\tilde{V}_{n-1}^n$  for particle  $i$  is denoted as  $X_{n,l}^i$ , and the MCMC mutation is presented jointly with resampling in step 3(c)ii.

---

**Algorithm 3.** An MCMC Procedure for  $\tilde{X}_{n,l}^i \sim \mathcal{K}_{n,l}^m(\cdot | X_{n,l}^i)$ .

---

- Initialize: Set  $V^{(0)} = X_{n,l}^i$ , and let  $W^{(0)} = W_{n,l}^i$  be the Wiener process generating  $X_{n,l}^i$ .
- For  $k = 1, \dots, m$ : Let  $V = V^{(k-1)}$ ,  $W = W^{(k-1)}$ .
  - Sample a new noise:

$$W(s)' = \rho W(s) + \sqrt{1 - \rho^2} \xi(s), \quad s \in (t_{n-1}, t_n], \quad \xi \sim \mathbb{W}.$$

- Obtain solution of SPDE (13) with  $W'$  the driving noise, i.e.,

$$dV'(s) = (-\nu AV'(s) - B(V'(s), V'(s))) dt + Q^{\frac{1}{2}} g(s, V'(s)) ds + Q^{\frac{1}{2}} dW'(s), \\ t \in (t_{n-1}, t_n].$$

- Compute acceptance ratio:

$$\alpha_{n,l} = 1 \wedge \frac{\left( \frac{dV_{n-1}^n}{d\mathbb{Q}_{n-1}^n}(V') p(Y_{t_n} | V'(t_n)) \right)^{\phi_{n,l}}}{\left( \frac{dV_{n-1}^n}{d\mathbb{Q}_{n-1}^n}(V) p(Y_{t_n} | V(t_n)) \right)^{\phi_{n,l}}}.$$

- With probability  $\alpha_{n,l}$  set  $V^{(k)} = V'$ ,  $W^{(k)} = W'$ ; otherwise, reject proposal, and set  $V^{(k)} = V$ ,  $W^{(k)} = W$ .
  - Return  $\tilde{X}_{n,l}^i = V^{(k)}$  and  $\tilde{W}_{n,l}^i = W^{(k)}$ .
-

**4.3.1. Extensions.** First, similarly with [17] one can extend the proposals by reducing the lower bound on the time we start adding noise (here  $t_{n-1}$ ). This could be made smaller, and this can be beneficial in terms of adding diversity, but for the sake of simplicity we do not pursue this further.

It is important to note that  $\mathcal{K}_{n,l}$  is based on adapting a very basic version of pCN-MCMC as outlined in [45, 11, 25]. There, typically  $\Lambda$  is chosen to be a Gaussian measure that coincides with a prechosen prior for a static Bayesian inference problem. The resulting MCMC kernel often exhibits slow mixing properties. This can be addressed by allowing a few selected coordinates to be proposed from a kernel invariant to a Gaussian approximation of the posterior distribution. The remaining coordinates are sampled as before (using kernels invariant to the prior), so that the scheme is valid for arbitrary dimensions. This results in more advanced pCN-MCMC algorithms with likelihood-informed proposals for  $\mathcal{Q}_{n,l}$  such as the ones described in [12, 34]. In the context of SMC one has the added benefit of using particle approximations for the mean and covariance to construct likelihood-informed proposals for  $\mathcal{Q}_{n,l}$ , and this results in a simple and effective approach, as illustrated in [31, 5].

A natural question to pose is how these ideas can be extended to construct more efficient  $\mathcal{K}_{n,l}$ . Note that the filtering problem is more complicated, as the variables of interest are SPDE paths. Still more advanced proposals can be implemented after a change of measure. For the MCMC above we chose  $\Lambda_n = \Pi_{n-1} \otimes \mathbb{Q}_{n-1}^n$ . This choice was because of its simplicity in implementation and its effectiveness in the numerical examples we considered, where the MCMC kernel in Algorithm 3 mixed well. When facing harder problems, one can extend the construction of  $\Lambda_n$  and use instead of  $\mathbb{Q}_{n-1}^n$  any measure that admits a Radon–Nikodym derivative w.r.t it. For example, one could use instead of (21) a proposal like

$$(22) \quad V(s)' = \rho V(s) + \sqrt{1 - \rho^2} W'(s), \quad t_{n-1} < s \leq t_n, \quad W' \sim \mathbb{W}$$

with  $\vartheta_{n,l} = \frac{d\Pi_{n,l}}{d(\Pi_{n-1} \otimes \mathbb{W}_{n-1}^n)}$ , where the Girsanov transformation between  $(t_{n-1}, t_n]$  can be established rigorously as in [7, Propositions 4.1 and 4.2]. This construction is an alternative to Algorithm 3, which is more amenable to extensions along the lines of [31, 5], as the reference measure is Gaussian. To follow [31, 5] one should use a Gaussian measure whose covariance operator should take into account the likelihood for low frequencies. This means one should use in (22) a different Gaussian measure than  $\mathbb{W}$ , which is identical to  $\mathbb{W}$  for high  $|k|$ , and for low  $|k|$  the diffusion constants are computed from particle approximations for the posterior mean and covariance (given  $\mathcal{Y}_n$ ) of a sequence  $(W_{t_i}; t_i = t_{n-1}, \dots, t_n)$  obtained just before the MCMC mutation.

**5. Numerical examples.** We solve SPDE (6) for  $\nu = 0.1$  and  $f = 0$  numerically using the exponential Euler scheme [28] for the finite-dimensional projection (8). For (8), we use a Fourier truncation with  $L = 64$ , i.e.,  $-64 \leq k_1, k_2 \leq 64$ . For  $\pi_0$  we use  $\beta = 0.5$ ,  $\alpha = 3$ , and  $\mu = v_0^\dagger$ , with  $v_0^\dagger$  being a random sample from  $\mathcal{N}(0, A^{-\alpha})$  that is also used as the true signal to generate the observations. To determine  $Q$  we use  $\sigma_k = \sqrt{2\delta\nu}|k|^{-3}$  with  $\delta = 1$ . For the observation equation in (10) we use  $\Sigma = 0.8I$ , and for the observer in (11) we place the observers' locations  $x_l$  on a uniform square grid with equal spacing and set  $r$  to be small (smaller than  $2\pi/L$ ). Thus, we can make the likelihood more informative by decreasing the observation noise or by increasing the grid size. As the information in the likelihood increases, one expects a larger number of tempering steps (and slower total execution times). When no tempering is used, this will lead to a much lower value for the ESS.

TABLE 1

Average results for number of tempering steps,  $ESS$ , and  $L^2$ -errors (with standard deviations in parentheses) obtained from 10 independent runs of each algorithm.  $IS$ -PF-T denotes using the guided proposal with tempering, and PF-T is bootstrap with tempering. The other two methods follow similarly and do not use tempering and MCMC steps. In all cases we use  $N = 100$ ,  $\delta t_n = 0.4$ . For the PF-T we use  $m = 20$  MCMC steps (in Algorithm 3) with  $\rho = 0.9$ , and for  $IS$ -PF-T we use  $m = 10$  and  $\rho = 0.5$ . For  $n = 1$  we also use a pCN proposal for  $V(0)$  that is invariant to  $\pi_0$  with the step sizes being  $\rho_0 = 0.98$  for PF-T and  $\rho_0 = 0.9$  for  $IS$ -PF-T.

$n =$	Tempering steps					$ESS$				
	1	2	3	4	5	1	2	3	4	5
IS-PF-T	5.6	4.7	4.4	4	4.3	64.87	73.88	63.02	57.01	53.03
PF-T	10.1	7.7	7.4	7.6	8.1	77.73	70.62	68.50	75.88	82.02
IS-PF	n/a					1.16	1.92	1.56	2.11	1.90
PF	n/a					1.00	1.00	1.01	1.13	1.06

$n =$	$L^2$ -errors				
	1	2	3	4	5
IS-PF-T	0.19 (0.0012)	0.26 (0.0002)	0.21 (0.0003)	0.16 (0.0001)	0.27 (0.0005)
PF-T	0.43 (0.0110)	0.32 (0.0029)	0.25 (0.0054)	0.23 (0.0023)	0.38 (0.0137)
IS-PF	0.31 (0.0033)	0.45 (0.0166)	0.42 (0.0062)	0.33 (0.0062)	0.46 (0.0023)
PF	0.85 (0.0185)	1.13 (0.1493)	0.86 (0.0810)	0.96 (0.0260)	1.15 (0.0467)
EnKF	0.66 (0.1151)	0.60 (0.0108)	0.65 (0.0194)	0.63 (0.0245)	0.74 (0.0138)

We present results from two types of experiments with simulated observations. In the first case we will look at a batch of  $n = 5$  observations from a dense grid ( $16 \times 16$ ). We use this short run to illustrate the efficiency and performance of the methodology. The length of the data set allows using multiple independent runs for the same observations. In the second experiment we use a large number of observations ( $n = 100$ ) obtained from an  $8 \times 8$  grid using both Gaussian and Student  $t$ -distributed additive noise. We show that the method performs well for the longer time and that performance is similar for both Gaussian and non-Gaussian observations.

We begin with the case of  $n = 5$  and dense observation grid ( $16 \times 16$ ). In Table 1 we present results for  $N = 100$  and  $\delta t_n = 0.4$  comparing a naive bootstrap PF, a PF that uses the informed proposal (13) for IS but without tempering (both based on Algorithm 1), a PF that uses tempering when sampling from the stochastic NSE dynamics in (6), and a PF that uses both tempering and (13) for IS. We show the number of tempering steps per batch of observations, the  $ESS$  at each observation time  $t_p$ , and  $L^2$ -errors between the true signal vorticity  $w^\dagger$  and the estimated posterior mean  $\hat{w}$  at each epoch, i.e.,

$$\int_{\Omega} \|\hat{w}(x, t_i) - w^\dagger(x, t_i)\|^2 dx.$$

For the  $L^2$ -errors we also include in Table 1 results from a standard ensemble Kalman filter (EnKF) [19]. It should be noted that the EnKF is computationally cheaper, and usually it is used with lower values for  $N$  than here. We include it not for the sake of a direct comparison but to provide a benchmark for performance.

When tempering is used, we present in Figure 1 selected typical estimated PDFs and scatter plots for a few chosen frequencies  $k$ 's. In the scatter plots the advantage of using (13) (in the bottom plot of Figure 1) results in higher dispersion of the particles relative to sampling from (6) (top plot). This is also apparent in the tails of the estimated PDFs. In Table 1 it is evident that when using tempering, IS resulted in about half of the tempering steps than when sampling from (6). In both cases, the

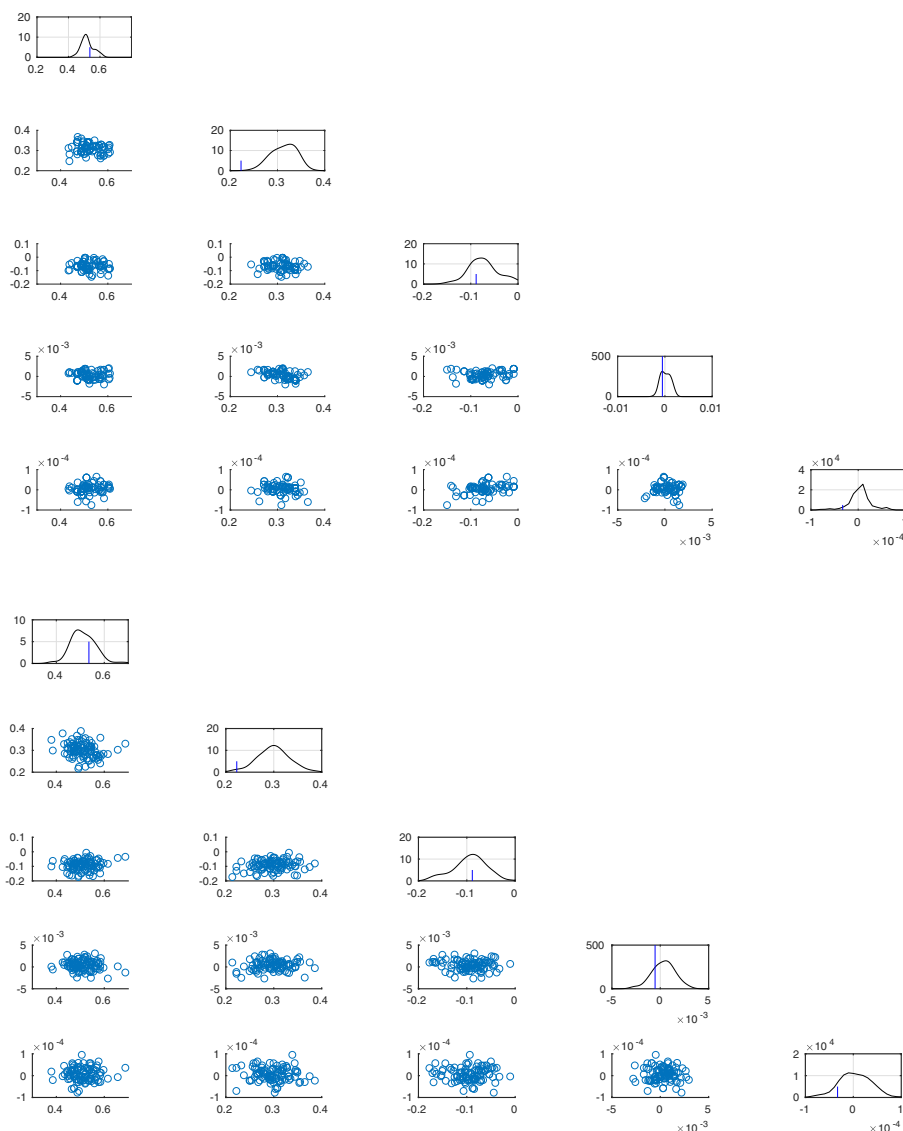


FIG. 1. PDF and scatter plots for  $\text{Real}(u_k)|\mathcal{Y}_n$  at  $n = 5$  for  $k = (1, 0), (1, 1), (1, -1), (2, 5), (9, 9)$ . Top is bootstrap (sampling with (6)), and bottom is IS (with (13)), and both use tempering. Vertical lines in PDF plots are true signal values used to generate the observations.

tuning of the MCMC steps lead to the same acceptance ratio (around 0.2 at the final tempering step). We use  $m = 20$  MCMC iterations per tempering for  $n = 1$ . For  $n > 1$ , plain tempering uses  $m = 20$ , and IS with (13) uses  $m = 10$ . In addition, the IS-tempering case uses a larger step size (smaller  $\rho$ ) for the MCMC (with  $\rho = 0.5$  rather than 0.9). This results in lower total computational cost and runtimes when IS is used despite the added computations imposed by computing  $g$  in (13). We also note that a lower number of tempering steps is beneficial in addressing potential path degeneracy issues. In Table 2 we present results when  $\delta t_n = 0.16, 1$  and  $\Sigma = 0.16I, 0.4I$  to illustrate the robustness of Algorithm 2 w.r.t spacing of observation times and signal to noise ratio. As expected, more tempering steps are needed in the more informative

TABLE 2

Average results of Algorithm 2 (IS-PF-T) when varying  $\Sigma$  and  $\delta t_n$ . Results are from 10 independent runs and presented similarly to Table 1. For the MCMC step sizes we use  $\rho_0 = 0.9$  (for  $\pi_0$ ) and  $\rho = 0.5, 0.5, 0.9, 0.5, 0.9$  for each case from top to bottom.

$n =$	Tempering steps					ESS				
	1	2	3	4	5	1	2	3	4	5
$\delta t_n = 0.4,$ $\Sigma = 0.8I$	5.6	4.7	4.4	4	4.3	64.87	73.88	63.02	57.01	53.03
$\delta t_n = 0.16,$ $\Sigma = 0.8I$	5.9	4.3	4.0	4.0	4.1	54.16	53.79	40.29	90.06	54.53
$\delta t_n = 1,$ $\Sigma = 0.8I$	5.1	4.6	4.6	4.6	5.1	65.25	45.23	80.27	99.78	82.45
$\delta t_n = 0.4,$ $\Sigma = 4I$	4.6	3.7	3.2	3.1	3.9	39.97	88.61	48.99	49.21	80.08
$\delta t_n = 0.4,$ $\Sigma = 0.16I$	7.2	6.0	5.9	5.7	6.1	65.27	74.57	51.94	54.57	50.26

$n =$	$L^2$ -errors				
	1	2	3	4	5
$\delta t_n = 0.4,$ $\Sigma = 0.8I$	0.19 (0.0012)	0.26 (0.0002)	0.21 0.0003)	0.16 (0.0001)	0.27 (0.0005)
$\delta t_n = 0.16,$ $\Sigma = 0.8I$	0.24 (0.0004)	0.22 (0.0018)	0.29 (0.0013)	0.26 (0.0019)	0.24 (0.0015)
$\delta t_n = 1,$ $\Sigma = 0.8I$	0.27 (0.0004)	0.28 (0.0005)	0.24 (0.0004)	0.15 (0.0001)	0.17 (0.00004)
$\delta t_n = 0.4,$ $\Sigma = 4I$	0.31 (0.0034)	0.61 (0.0057)	0.50 (0.0027)	0.38 (0.0021)	0.56 (0.0009)
$\delta t_n = 0.4,$ $\Sigma = 0.16I$	0.12 (0.0003)	0.10 (0.00009)	0.08 (0.00006)	0.07 (0.00001)	0.12 (0.0001)

observation case ( $\Sigma = 0.16I$ ), but at the same time accurate observations result in lower  $L^2$ -errors. In addition, our method seems to perform comparatively better when  $\delta t_n = 1$ . This can be attributed to the guided proposal being given more time to evolve and guide the particles to better regions of the state space.

We proceed with the second numerical experiment, where we use only a single run of a PF with both tempering and IS for  $N = 100$  and  $n = 1, \dots, 100$ . The dynamics for the state and true signal are as before, but for the observations we use an  $8 \times 8$  equally spaced observation grid and look at two different generated data sets with different distributions for the noise  $\Xi_n$ : a zero mean Gaussian and zero mean Student  $t$  distribution with 4 degrees of freedom. In both cases  $\Sigma = 0.8I$ . For each case, different PFs are implemented, each with the correctly specified likelihood. In Figure 2 we plot the estimated vorticity posterior mean for  $n = 10, 50, 100$  in each case together with the vorticity of the true signal. The estimates seem accurate with small deviations between the posterior mean and the true signal. The latter is sensible given the coarseness of the grid and the moderate number of observations. We also provide in Figure 3 a plot of the ratio of the posterior variance of the vorticity of  $V_{t_n}$  over the unconditional variance when obeying the probability law determined by the stochastic NSE dynamics in (6). The information gain appears as a reduction in the posterior variance for low  $|k|$  relative to the prior, which is to be expected, as the spatial grid cannot be informative for higher wavenumbers. In Figure 4 we plot  $ESS$ ,  $L^2$ -errors as before, and number of tempering steps per iteration. In both cases the performance is fairly stable with time, and the algorithm provides good posterior mean estimates. For completeness, in Figure 5 we include a comparison with the EnKF in terms of  $L^2$ -errors. The PF with IS and tempering performs much better. Finally, in Figure 6 we present some estimated PDFs. These plots capture  $\Pi_n$  for



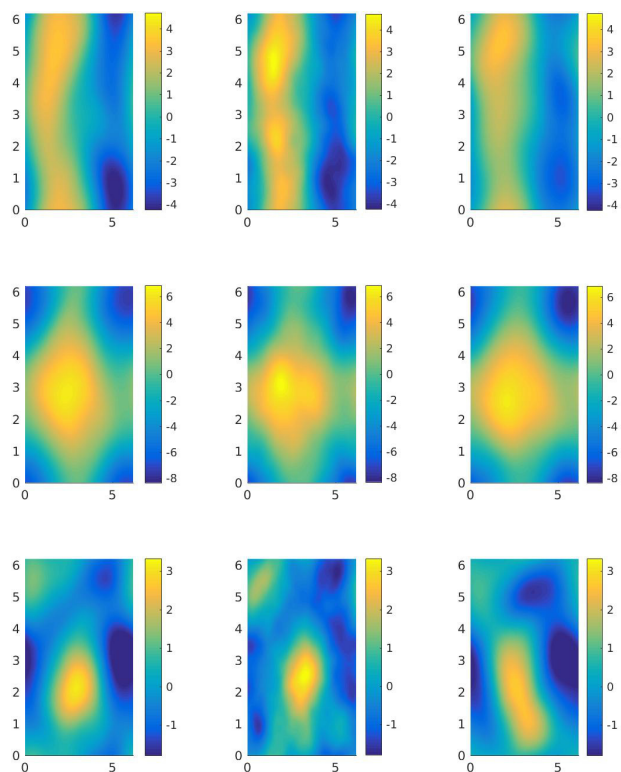


FIG. 2. Vorticity plots showing posterior mean of  $p(\nabla \times V_t | \mathcal{Y}_n)$  and true signal: top row  $n = 10$ , middle  $n = 50$ , bottom  $n = 100$ . The left column contains posterior means from Gaussian observation noise and the right one from Student  $t$  noise, and in the middle is  $w_t^\dagger$  (true signal vorticity).

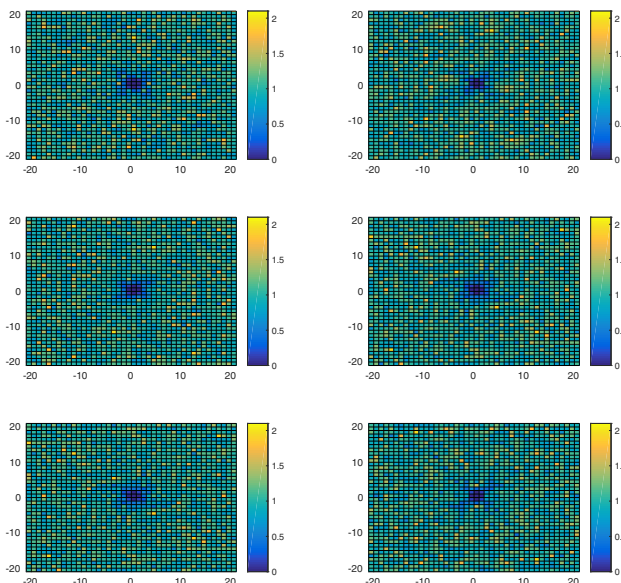


FIG. 3. Variance plots: top row  $n = 10$ , middle  $n = 50$ , bottom  $n = 100$ . We present heat maps of the ratio of the posterior variance of  $\pi_{t_n}$  over the variance for the law of the signal dynamics against  $k$ ; left part is for Gaussian noise and right for Student  $t$ .

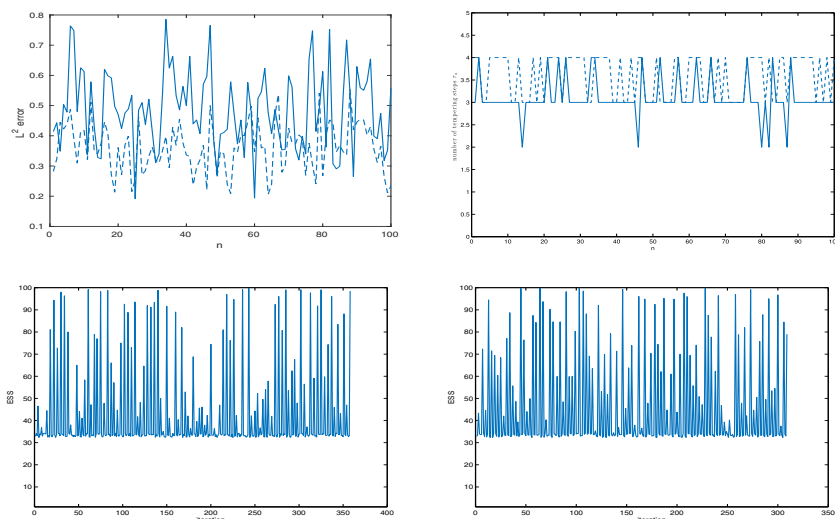


FIG. 4. Results for single run of PF with tempering and IS for  $8 \times 8$  grid. Top panels are  $L^2$ -errors (left) and number of tempering steps against  $n$  (right). Dotted lines are for Gaussian observation noise and solid for Student  $t$ . In the bottom panels we present ESS against SMC iteration for Gaussian (left) and Student  $t$  (right) errors. Execution times were  $4.8018 \times 10^5$  and  $4.17196 \times 10^5$  seconds, respectively.

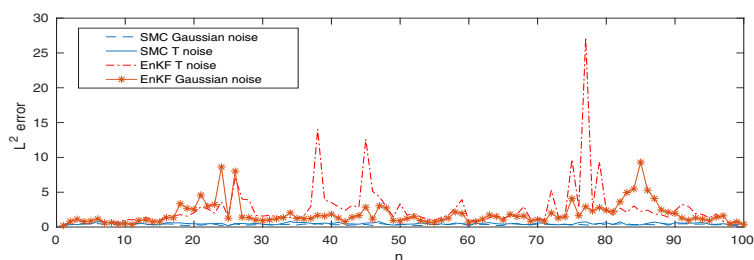


FIG. 5.  $L^2$ -error comparison for IS-PF with tempering and EnKF with Gaussian and Student  $t$  observation noise. PF errors are same as top left panel of Figure 4 and much lower than EnKF.

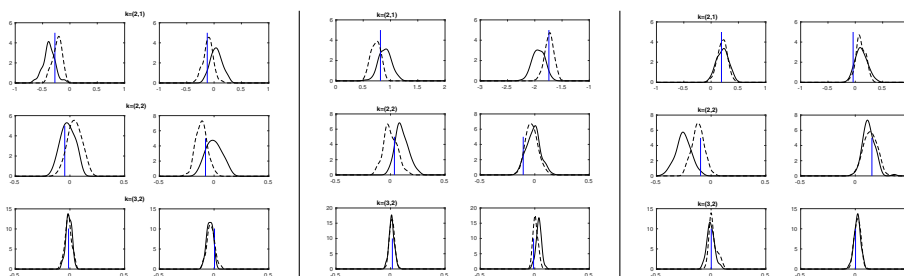


FIG. 6. PDFs of vorticity for  $n = 10, 50, 100$  from left to right. In each panel for  $n$ , left side displays real part, and right is imaginary; top row is  $k = (2, 1)$ , middle is  $k = (2, 2)$  and bottom is  $k = (3, 2)$ . Dotted line is for Gaussian observation noise and solid for Student  $t$ , and vertical lines are true signal values used to generate the observations.

different  $k$ . Notice that the true parameter (displayed as a vertical line) lies in regions where the mass of the estimated posterior density is high and the posterior variance for the  $t$ -distributed case is higher for low  $k$ .

**6. Discussion.** We have presented a particle filtering methodology that uses likelihood-informed IS proposals, tempering, and MCMC moves for signals obeying the stochastic NSE observed with additive noise. The approach is computationally intensive and requires a significant number of particles  $N$ , but we believe the cost is quite moderate relative to the dimensionality of the problem. The use of tempering and MCMC steps is crucial for this high-dimensional application. The inclusion of likelihood-informed proposals results in higher efficiency and *ESS*, less tempering steps, and higher step sizes for the MCMC steps and thus, overall, in lower computational cost. The IS proposals were designed using a Gaussian noise assumption for the observations, but we demonstrated numerically that they are still useful and efficient for observation noise obeying a Student  $t$  distribution with heavier tails. In addition, as  $\delta t_n$  increases, using proposals as in (13) will be more beneficial.

In the experiments presented in section 5 the effective dimensionality of the problem is determined by  $\nu$ ,  $\Sigma$  and  $\sigma_k$ . More challenging parameterizations than the ones presented here could be dealt with by increasing  $N$  or via a more advanced numerical method for the solution of the SPDE. These can be addressed using the extensions discussed in section 4.3.1. Another potentially useful extension is to use different number of particles for different ranges of  $k$  following [30]. Furthermore, we note that we did not make use of parallelization, but this is certainly possible for many parts of Algorithm 2 and can bring significant execution speedups in applications.

Future work could aim to extend this methodology by designing suitable IS proposals for nonlinear observation schemes or observations obtained from Lagrangian drifters or floaters. Finally, an interesting question is whether an error analysis along the lines of [14, section 7.4] can be reproduced. The simulations presented here seem to indicate roughly constant errors with time, but a rigorous treatment would need to establish the stability properties of the filtering distribution w.r.t the initialization.

**Appendix A. More simulation results.** We present some negative numerical results to illustrate that tempering is necessary. We will consider a perfect initialization for each particle with  $v_0^\dagger$ . While this is an extremely favorable scenario that is unrealistic in practice, it shows a clear benefit in using IS and tempering. For  $N = 200$ ,  $\nu = 0.01$ , and  $\delta t_n = 0.2$ , we present some scatter plots in Figure 7 for the experiment with  $n = 5$  seen earlier with a  $16 \times 16$  block of observations. Notably, the estimated posterior means for the vorticity seem to exhibit good performance; see Figure 8. Indicatively, the ESS here is 34 for IS and 3 for the bootstrap case. Even in this extremely favorable scenario, the ESS is low, and this strongly motivates the use of tempering to improve the efficiency of the particle methodology. In results not shown here, we have also experimented with the size of the time increment  $\delta t_n = t_n - t_{n-1}$  for what a naïve particle filter (Algorithm 1) can handle. When the likelihood-informed proposals in (13) are used, the method produces accurate point estimates for  $\delta t_n$  up to  $0.2 - 0.25$ . This is in contrast to when sampling from the dynamics, where the bootstrap version of Algorithm 1 can handle only up to 0.15.

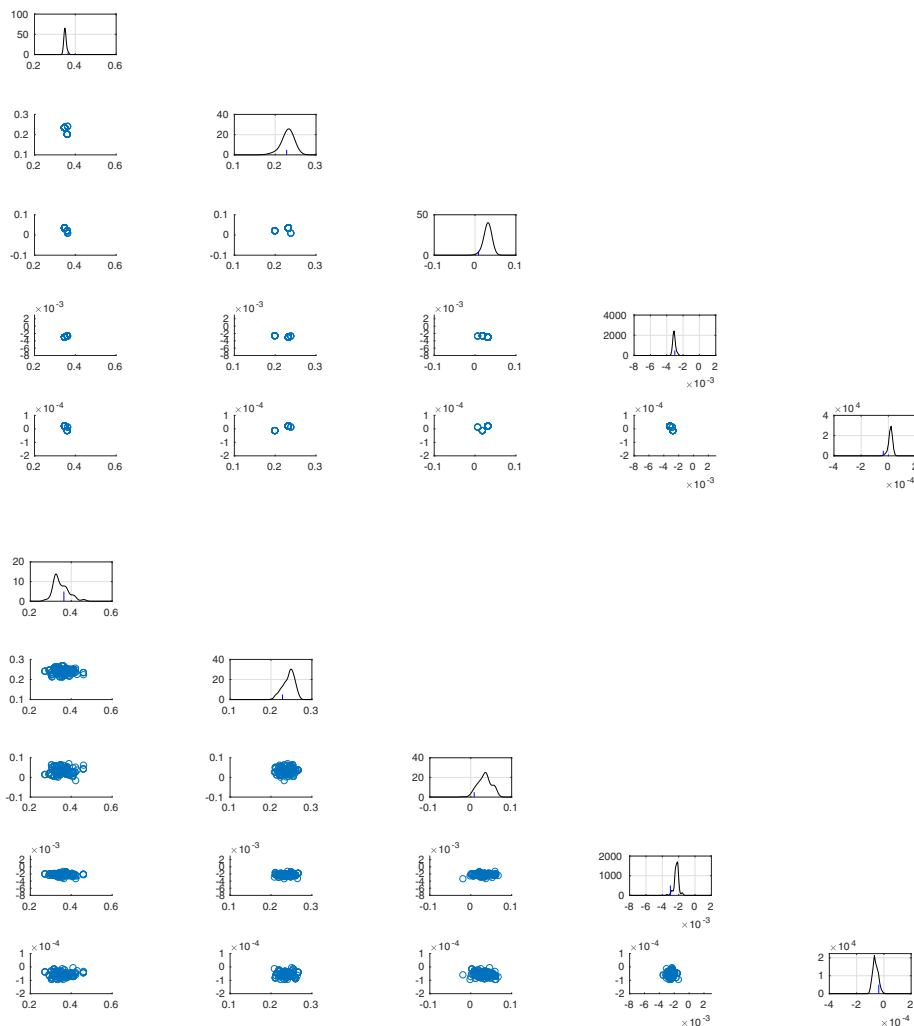


FIG. 7. Scatter plots at  $n = 5$  for perfect initialization  $k = (1, 0), (1, 1), (1, -1), (2, 5), (9, 9)$ . Top is bootstrap, and bottom is IS with (13).

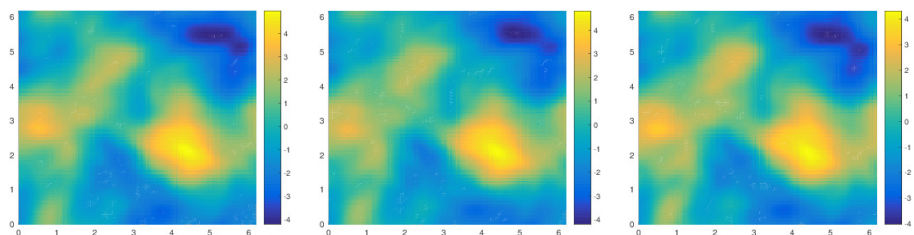


FIG. 8. Vorticity plots for  $n = 5$  and perfect initialization: Left is posterior mean from bootstrap PF, middle is real signal  $v_t^\dagger$ , and right is posterior mean of PF of Algorithm 1 and IS with (13).

**Acknowledgments.** A.J. is affiliated with the Risk Management Institute, the Center for Quantitative Finance, and the OR & Analytics cluster at NUS.

## REFERENCES

- [1] A. BAIN AND D. CRISAN, *Fundamentals of Stochastic Filtering*, Springer, New York, 2008.
- [2] A. F. BENNETT, *Inverse Modeling of the Ocean and Atmosphere*, Cambridge University Press, Cambridge, 2005.
- [3] A. BESKOS, D. CRISAN, A. JASRA, K. KAMATANI, AND Y. ZHOU, *A stable particle filter for a class of high-dimensional state-space models*, Adv. Appl. Probab., 49 (2017), pp. 24–48.
- [4] A. BESKOS, A. JASRA, N. KANTAS, AND A. THIERY, *On the convergence of adaptive sequential Monte Carlo methods*, Ann. Appl. Probab., 26 (2016), pp. 1111–1146.
- [5] A. BESKOS, A. JASRA, K. LAW, Y. MARZOUK, AND Y. ZHOU, *Multilevel sequential Monte Carlo with dimension-independent likelihood-informed proposals*, SIAM/ASA J. Uncertain. Quantif., to appear.
- [6] M. BOCQUET, C. A. PIRES, AND L. WU, *Beyond Gaussian statistical modeling in geophysical data assimilation*, Monthly Weather Revi., 138 (2010), pp. 2997–3023.
- [7] M.-H. CHANG, *Large deviation for Navier-Stokes equations with small stochastic perturbation*, Appl. Math. Comput., 76 (1996), pp. 65–93.
- [8] N. CHOPIN, *A sequential particle filter method for static models*, Biometrika, 89 (2002), pp. 539–552.
- [9] A. J. CHORIN AND P. KRAUSE, *Dimensional reduction for a Bayesian filter*, Proc. Natl. Acad. Sci. USA, 101 (2004), pp. 15013–15017.
- [10] A. J. CHORIN, M. MORZFELD, AND X. TU, *Implicit particle filters for data assimilation*, Comm. Appl. Math. Comput. Sci., 5 (2010), pp. 221–240.
- [11] S. L. COTTER, G. O. ROBERTS, A. M. STUART, AND D. WHITE, *MCMC methods for functions: modifying old algorithms to make them faster*, Stat. Sci., 28 (2013), pp. 424–446.
- [12] T. CUI, K. J. LAW, AND Y. M. MARZOUK, *Dimension-independent likelihood-informed MCMC*, J. Comput. Phys., 304 (2016), pp. 109–137.
- [13] G. DA PRATO AND J. ZABCZYK, *Stochastic Equations in Infinite Dimensions*, Cambridge University Press, Cambridge, 2008.
- [14] P. DEL MORAL, *Feynman-Kac Formulae*, Springer, New York, 2004.
- [15] P. DEL MORAL, *Mean Field Simulation for Monte Carlo Integration*, CRC Press, Boca Raton, FL, 2013.
- [16] P. DEL MORAL, A. DOUCET, AND A. JASRA, *Sequential Monte Carlo samplers*, J. Roy. Statist. Soc. Ser. B, 68 (2006), pp. 411–436.
- [17] A. DOUCET, M. BRIERS, AND S. SÉNÉCAL, *Efficient block sampling strategies for sequential Monte Carlo methods*, J. Comput. Graph. Statist., 15 (2006), pp. 693–711.
- [18] A. DOUCET, N. DE FREITAS, AND N. GORDON, *Sequential Monte Carlo Methods in Practice*, Springer Science & Business Media, New York, 2001.
- [19] G. EVENSEN, *Data Assimilation: The Ensemble Kalman Filter*, Springer, New York, 2009.
- [20] B. FERRARIO, *Stochastic Navier-Stokes equations: Analysis of the noise to have a unique invariant measure*, Ann. Mat. Pura Appl., 177 (1999), pp. 331–347.
- [21] F. FLANDOLI, *Dissipativity and invariant measures for stochastic Navier-Stokes equations*, No DEA Nonlinear Differential Equations Appl., 1 (1994), pp. 403–423.
- [22] F. GIRAUD AND P. DEL MORAL, *Nonasymptotic analysis of adaptive and annealed Feynman-Kac particle models*, Bernoulli, 23 (2017), pp. 670–709.
- [23] S. GODSILL AND T. CLAPP, *Improvement strategies for Monte Carlo particle filters*, in Sequential Monte Carlo Methods in Practice, A. Doucet, N. de Freitas, and G. Gordon, eds., Springer, New York, 2001, pp. 139–158.
- [24] A. GOLIGHTLY AND D. J. WILKINSON, *Bayesian inference for nonlinear multivariate diffusion models observed with error*, Comput. Statist. Data Anal., 52 (2008), pp. 1674–1693.
- [25] V. H. HOANG, K. J. LAW, AND A. M. STUART, *Determining white noise forcing from Eulerian observations in the Navier-Stokes equation*, Stoch. Partial Differ. Equ. Anal. Comput., 2 (2014), pp. 233–261.
- [26] M. JARDAK, I. NAVON, AND M. ZUPANSKI, *Comparison of sequential data assimilation methods for the Kuramoto-Sivashinsky equation*, Int. J. Numer. Methods Fluids, 62 (2010), pp. 374–402.
- [27] A. JASRA, D. A. STEPHENS, A. DOUCET, AND T. TSAGARIS, *Inference for Lévy-driven stochastic volatility models via adaptive sequential Monte Carlo*, Scand. J. Statist., 38 (2011), pp. 1–22.

- [28] A. JENTZEN AND P. E. KLOEDEN, *Overcoming the order barrier in the numerical approximation of stochastic partial differential equations with additive space-time noise*, Proc. Roy. Soci. Lond. A, 465 (2009), pp. 649–667.
- [29] A. M. JOHANSEN, *On block, tempering and particle mcmc for systems identification*, in Proceedings of 17th IFAC Symposium on System Identification, IFAC, Laxenburg, Austria, 1998.
- [30] A. M. JOHANSEN, N. WHITELEY, AND A. DOUCET, *Exact approximation of Rao–Blackwellised particle filters*, IFAC Proc. Vol., 45 (2012), pp. 488–493.
- [31] N. KANTAS, A. BESKOS, AND A. JASRA, *Sequential Monte Carlo methods for high-dimensional inverse problems: A case study for the Navier-Stokes equations*, SIAM/ASA J. Uncertain. Quantif., 2 (2014), pp. 464–489.
- [32] S. KUKSIN AND A. SHIRIKYAN, *Mathematics of Two-Dimensional Turbulence*, Vol. 194, Cambridge University Press, Cambridge, 2012.
- [33] K. LAW, A. STUART, AND K. ZYGALAKIS, *Data Assimilation: A Mathematical Introduction*, Vol. 62, Springer, New York, 2015.
- [34] K. J. LAW, *Proposals which speed up function-space MCMC*, J. Comput. Appl. Math., 262 (2014), pp. 127–138.
- [35] A. LEE, C. YAU, M. B. GILES, A. DOUCET, AND C. C. HOLMES, *On the utility of graphics cards to perform massively parallel simulation of advanced Monte Carlo methods*, J. Comput. Graph. Statist., 19 (2010), pp. 769–789.
- [36] A. J. MAJDA AND J. HARLIM, *Filtering Complex Turbulent Systems*, Cambridge University Press, Cambridge, 2012.
- [37] R. M. NEAL, *Annealed importance sampling*, Stat. Comput., 11 (2001), pp. 125–139.
- [38] N. OUDJANE AND C. MUSSO, *Progressive correction for regularized particle filters*, in Information Fusion, 2000, Proceedings of the Third International Conference on FUSION 2000, Vol. 2, IEEE, New York, 2000, pp. THB2–THB10.
- [39] N. PAPADAKIS, É. MÉMIN, A. CUZOL, AND N. GENGEMBRE, *Data assimilation with the weighted ensemble Kalman filter*, Tellus A, 62 (2010), pp. 673–697.
- [40] O. PAPASPILIOPOULOS AND G. ROBERTS, *Importance sampling techniques for estimation of diffusion models*, Stat. Methods Stochastic Differential Equations, 124 (2012), pp. 311–340.
- [41] O. PAPASPILIOPOULOS, G. O. ROBERTS, AND O. STRAMER, *Data augmentation for diffusions*, J. Comput. Graph. Statist., 22 (2013), pp. 665–688.
- [42] S. SÄRKKÄ AND E. MOULINES, *On the  $L_p$ -convergence of a Girsanov theorem based particle filter*, in 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, New York, 2016, pp. 3989–3993.
- [43] S. SÄRKKÄ AND T. SOTTINEN, *Application of Girsanov theorem to particle filtering of discretely observed continuous-time nonlinear systems*, Bayesian Anal., 3 (2008), pp. 555–584.
- [44] C. SNYDER, T. BENGTTSSON, P. BICKEL, AND J. ANDERSON, *Obstacles to high-dimensional particle filtering*, Monthly Weather Rev., 136 (2008), pp. 4629–4640.
- [45] A. M. STUART, *Inverse problems: A Bayesian perspective*, Acta Numer., 19 (2010), pp. 451–559.
- [46] L. TIERNEY, *A note on Metropolis-Hastings kernels for general state spaces*, Ann. Appl. Probab., (1998), pp. 1–9.
- [47] F. VAN DER MEULEN AND M. SCHAUER, *Bayesian Estimation of Incompletely Observed Diffusions*, preprint, arXiv:1606.04082, 2016.
- [48] P. J. VAN LEEUWEN, *Nonlinear data assimilation in geosciences: An extremely efficient particle filter*, Quart. J. Roy. Meteorol. Soc., 136 (2010), pp. 1991–1999.
- [49] P. J. VAN LEEUWEN, Y. CHENG, AND S. REICH, *Nonlinear Data Assimilation*, Springer, New York, 2015.
- [50] J. WEARE, *Particle filtering with path sampling and an application to a bimodal ocean current model*, J. Comput. Phys., 228 (2009), pp. 4312–4331.
- [51] G. A. WHITAKER, A. GOLIGHTLY, R. J. BOYS, AND C. SHERLOCK, *Improved bridge constructs for stochastic differential equations*, Stat. Comput., (2016), pp. 1–16.